

A Multiagent Simulation for Traffic Flow Management with Evolutionary Optimization

Patryk Filipiak
Institute of Computer Science
University of Wrocław, Poland
patryk.filipiak@ii.uni.wroc.pl

August 18, 2011

Abstract

A traffic flow is one of the main transportation issues in nowadays industrialized agglomerations. Configuration of traffic lights is among the key aspects in traffic flow management. This paper proposes an evolutionary optimization tool that utilizes multiagent simulator in order to obtain accurate model. Even though more detailed studies are still necessary, a preliminary research gives an expectation for promising results.

1 Traffic flow management

Early models of traffic flow (called *macroscopic* models) treated vehicles in the collective manner basing on the analogy to particles in a fluid [7]. Later on, more precise (*mesoscopic*) models were being created consecutively (mostly based on gas kinetics) [5]. Currently, a multiagent simulation mechanism provides much more efficient *microscopic* model where each vehicle can be regarded separately allowing for highly detailed analysis including collision avoidance [4], traffic virtualization [2], interactions with pedestrians [3], etc.

Traffic flow management varies from tracing main roads average capacity across certain area (where macroscopic models give satisfactory results) to very low-level manipulations including re-arrangement of lanes, modifying traffic lights configuration, planning bridge locations, etc. where microscopic models are most suitable [9].

2 Multiagent traffic flow simulator

The agent-based traffic flow simulator described in [9] is the universal microscopic model. The environment for agents in this model comprises of a system of city streets defined in XML files using the following entities illustrated in Figure 1:

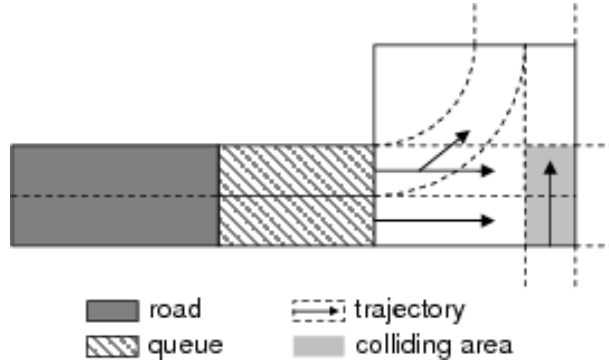


Figure 1: Elements of environment description presented on the part of a sample junction.

- **road** – defines the one-way part of a street, i.e. an existing path leading from one junction to another. For the sake of simplicity, roads at the frontiers of a model are considered dead-end even though they play roles of ingoing or outgoing roads.
- **queue** – defines a single lane represented as a first-in-first-out list of vehicles located on this lane one after another. Note that there has to be at least one queue for each road at each junction.
- **trajectory** – corresponds to a trajectory of vehicles crossing the junction from a given ingoing road to a given outgoing road. Each trajectory stores the information about other trajectories with possible collisions with it.
- **track** – an abstract pair *queue-trajectory* binding three entities, namely queue at ingoing road, queue at outgoing road and trajectory from one to another.

Each vehicle in the environment is an agent that is aware of its current speed and exact location within a model (particularly, its relative position on the present road). What is more, a vehicle is assigned another agent who drives it in order to achieve the driver's own aim which is to get from point A to point B choosing the best possible way.

Driver agents are equipped with a simple perception mechanism that allows them to acquire information about a vehicle driven by them (e.g. current lane, driving speed or acceleration) and the current state of environment in their proximity, particularly: other vehicles (their behaviour, distance to them, etc.) and traffic lights.

For the sake of efficiency, described model includes the following simplifications that should be taken into account: there are traffic lights at each junction, there are no roundabouts and the presence of pedestrians is ignored.

Simulation process in described model is the sequence of $N \in \mathbb{N}$ iterations executed every $\tau > 0$ milliseconds (by default $N = 1000$ and $\tau = 200$). Each iteration includes the following operations:

1. Apply changes in the environment (if there are any at this time period) including the presence of colliding tracks and the state of traffic lights.
2. Refresh all agents' local information about the state of environment.
3. Refresh locations of all moving vehicles according to intentions of their drivers considering current speed and acceleration of vehicles.
4. Remove agents that achieved their aims and create new ones if possible.

Vehicle agents collect some essential information during simulation, i.e. exact trace; number, locations and durations of stops; average speed etc. for the sake of further analysis.

Figure 3 depicts arrangement of main streets in Wroclaw, Poland that can be easily modelled using described agent-based simulator.

3 Optimization of traffic lights

Traffic lights are commonly used as an effective solution for car flow routine regulation in virtually all agglomerations. Modification of traffic lights configuration is the key aspect in optimization of traffic flow for the model presented in the previous section.

In many real-world situations, traffic lights configuration settings are based on combination of averaged theoretical data and designer's assumptions. A multiagent simulation approach can be used instead in order to provide more accurate data.

A cyclic sequence of consecutive changes of lights including detailed information about duration of time when each light is on combined with dependencies amongst all traffic lights within a given area (typically a crossroad) is referred to as *traffic lights programme*. The usual representation of such programme is a graphical timeline presented in Figure 2.

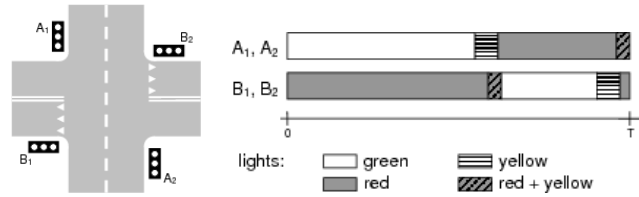


Figure 2: A sample junction with two pairs of traffic lights (left) and a corresponding traffic lights programme (right).



Figure 3: Arrangement of main streets in Wrocław (Poland) that can be easily modelled with the agent-based simulator.

The merit of underlying optimization process lies in realistic accuracy provided by the agent-based simulator. Performance of contemporary hardware is sufficiently enough for launching simulations containing even millions of agents in place of less accurate statistical estimations that were used in previous years. Thus, the values of a function that is being optimized are in fact results of agent-based simulation.

3.1 Constraints

Any traffic lights programme is obliged to satisfy the following constraints:

- A change of lights cannot violate the sequence (green, yellow, red, red+yellow) repeated continuously.
- Duration of green light must be greater or equal to a given minimal time length whereas durations of transient lights (i.e. yellow and red+yellow) are constant and cannot undergo any optimization.
- No collision possibilities are allowed, i.e. red light must be lit on at one pair of traffic lights until green light turns to yellow on another pair and remains yellow for a given time period; similar condition must be satisfied during a change from green to red light.

3.2 Criteria

The aim is to optimize a traffic lights programme by maximizing a car flow subject to constraints described above. Typical optimization criteria in such case would be: maximizing the number of agents that achieved their aims during simulation, minimizing the average time of a ride, maximizing average speed of vehicles, etc. Since most of the criteria correspond to each other an aggregation of at least two of them provides a fair evaluation function.

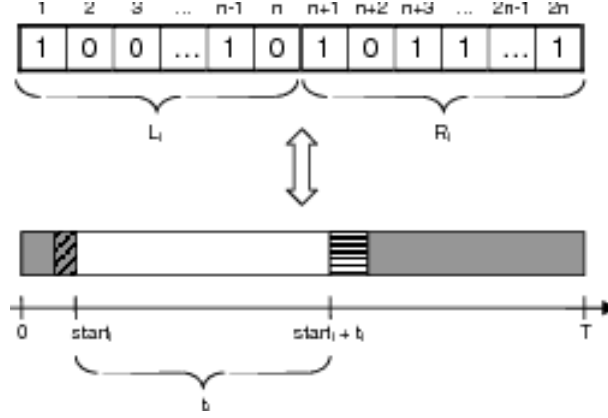


Figure 4: A sample programme of i -th traffic lights represented in binary chromosome form (up) and its corresponding timeline form (down).

3.3 Representation

Assume that $T > 0$ is a time length of one traffic lights cycle at a junction. Time range $[0, T]$ can be discretized into T/τ intervals of $\tau > 0$ milliseconds each as it was mentioned in the previous section. Let t_{min} be the minimal and

$$t_{max} = \frac{T}{\tau} - k \cdot t_{min} - C, \quad (C = \text{const}) \quad (1)$$

be the maximal feasible time length of green light duration. Note that t_{max} is determined by the time length of green light duration for k tracks colliding with current track and some constant value reserved for transient lights.

As a result, any programme of $M > 0$ traffic lights can be well-defined by the set of pairs

$$\{(start_1, t_1), (start_2, t_2), \dots, (start_M, t_M)\}, \quad (2)$$

where $start_i$ represents the number of time interval when i -th traffic light turns to green and remains for the next t_i time intervals ($i = 1, \dots, M$). Duration of red light can be computed therefore as $T - t_i - C$.

Evolutionary algorithms provide an efficient optimization mechanism for the problem presented above. Let $n > 0$ be the lowest integer value that satisfies condition $T/\tau \leq 2^n$, hence

$$0 < t_{min} \leq t_{max} \leq 2^n. \quad (3)$$

Any traffic lights programme expressed in the form (2) can be encoded as a $2MN$ -element binary chromosome containing M pairs of N -bit values (L_i, R_i) such that

$$start_i = L_i \bmod T/\tau, \quad (4)$$

$$t_i = t_{min} + [R_i \bmod (t_{max} - t_{min} + 1)] \quad (5)$$

for $i = 1, \dots, M$.

3.4 Algorithm

Population-Based Incremental Learning (PBIL) [1] can be used as an evolutionary optimization tool for the stated problem thanks to its simplicity and promising results in a broad range of applications. It is also worth noticing that even more complex evolutionary algorithms could be applied instead without a significant decrease of performance due to considerably time consuming evaluation process that is suggested in this paper.

Algorithm 1 presents the pseudocode of PBIL. It is so called Estimation of Distribution Algorithm (EDA) [6] based on the idea of optimizing a probability model for solution rather than implicit set of solutions. Later on, individuals are randomly generated according to obtained probability distribution.

The requirement for collision avoidance is not satisfied by the traffic lights programme definition itself, hence the presence of infeasible individuals in population might be expected. Figure 5 presents a simple conflicts resolving method for this case. If a collision possibility occurs between a pair of trajectories then corresponding traffic lights programme can be modified in order to ensure arbitrary set C' distance between green light durations. If any of modified green light durations becomes shorter than t_{min} then it is automatically expanded to t_{min} in the only possible way. Finally, if a conflict still occurs, a new random individual is selected.

Algorithm 1 Standard PBIL algorithm with parameters $0 < \theta_1, \theta_2, \theta_3 < 1$ on a population of *size* individuals each of which is represented as d -element binary vector.

```

 $p \leftarrow \text{InitialProbabilityVector}()$ 
 $P \leftarrow \text{RandomPopulation}(p, \text{size})$ 
 $\text{PopulationEvaluation}(P)$ 
while not  $\text{TerminationCondition}(P)$  do
   $x_i \leftarrow \text{BestIndividual}(P)$ 
  for  $k \leftarrow 1$  to  $d$  do
     $p_k \leftarrow p_k \cdot (1 - \theta_1) + x_{ik} \cdot \theta_1$ 
  end for
  for  $k \leftarrow 1$  to  $d$  do
    if  $\text{UniformRandom}(0, 1) \leq \theta_2$  then
       $p_k \leftarrow p_k \cdot (1 - \theta_3) + \text{BinaryRandom}(0.5) \cdot \theta_3$ 
    end if
  end for
   $P \leftarrow \text{RandomPopulation}(p, \text{size})$ 
   $\text{PopulationEvaluation}(P)$ 
end while

```

4 Conclusions and Perspectives

Even though the results are partial at the moment and more detailed studies are still necessary, a preliminary research gives an expectation for promising results. An improvement of agent-based traffic flow model could be a next step. Also a modification of evolutionary mechanism may improve the performance of optimization process. Replacing PBIL with Bayesian Optimization Algorithm (BOA) [8] could ensure obtaining much more detailed model. Unlike other evolutionary algorithms, BOA not only searches for optimal solution but also gives some information about its structure. Lots of dependencies (e.g. among traffic lanes and even adjacent junctions) could be modelled in terms of Bayesian networks that are utilized in BOA. What is more, such dependency model could include the fact that more effort should be put to traffic flow management inside city centers than suburbs because of greater amount of junctions, narrow and/or one-way street, etc.

As it is clearly seen, the effectiveness of agent-based approach applied for traffic flow managements provides a broad range of ideas.

5 Acknowledgements

The multiagent simulator developed by Krzysztof Sroka within confines of his Master's dissertation [9] was extensively used during research process for this paper.

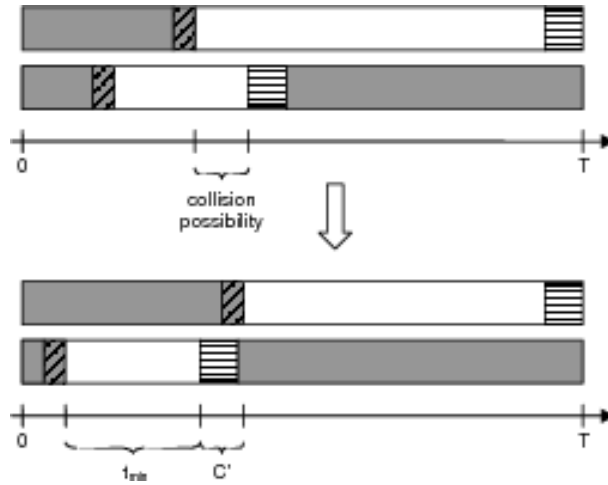


Figure 5: Simple conflicts resolving method.

References

- [1] Baluja, S., *Population-Based Incremental Learning: A Method for Integrating Genetic Search Based Function Optimization and Competitive Learning*, Technical Report (Pittsburgh, PA: Carnegie Mellon University), 1994.
- [2] Berg van den, J., Sewall, J., Lin, M., Manocha, D., *Virtualized Traffic: Reconstructing Traffic Flows from Discrete Spatio-Temporal Data* in Proceedings of IEEE VR, 2009.
- [3] Godara, A., Lassarre, S., Banos, A., *Simulating Pedestrian-Vehicle Interaction in an Urban Network Using Cellular Automata and Multi-Agent Models*. Lecture Notes in Computer Science, Traffic and Granular Flow05, Part IV, 2007
- [4] Hoffmann, G., M., Tomlin, C., J., *Decentralized cooperative collision avoidance for acceleration constrained vehicles*. 47th IEEE Conference on Decision and Control (CDC), 2008.
- [5] Hoogendoorn, S., P., Bovy, P., H., L., *State-of-the-art of Vehicular Traffic Flow Modelling*, Special Issue on Road Traffic Modelling and Control of the Journal of Systems and Control Engineering, 2001.
- [6] Larranaga, P., Lozano, J. A., *Estimation of distribution algorithms: A new tool for evolutionary computation*. Kluwer Academic Publishers, Boston, 2002.
- [7] Lighthill, M., H., Whitham, G., B., *On kinematic waves II: a theory of traffic flow on long, crowded roads*. Proceedings of the Royal Society of London series A, 1955.
- [8] Pelikan, M., Goldberg, D. E., Cantu-Paz, E., *Linkage problem, distribution estimation, and Bayesian networks*. IlliGAL Report No. 98013: Illinois Genetic Algorithm Laboratory, University of Illinois at Urbana-Champaign, 1998.
- [9] Sroka, K., *A traffic lights programme optimization*, Master's Thesis, Institute of Computer Science, University of Wroclaw, 2009.